The Delphion Integrated View

Title: **JP2001014201A2: SHARED FILE SYSTEM**

Country: **JP** Japan

Kind: **A2** Document Laid open to Public inspection !

Inventor: **YAMAMOTO TOSHIYUKI**;

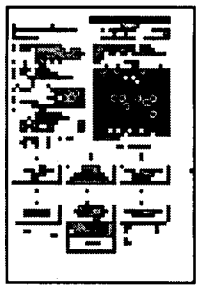
Assignee: **TOSHIBA CORP**
News, Profiles, Stocks and More about this company

Published / Filed: **2001-01-19** / 1999-06-29

Application Number: **JP1999000183218**

IPC Code: **G06F 12/00;** G06F 3/06; G06F 11/00; G06F 15/177;

Priority Number: 1999-06-29 **JP1999000183218**



View Image

1 page

Abstract:

PROBLEM TO BE SOLVED: To prevent a file system from being destroyed even though a split brain takes place.

SOLUTION: Each server 102 is provided with a distributed controller 202, a file operation module 201 and a heartbeat adapter 204. The controller 202 outputs the address of a free area on a shared disk drive 101 to the module 201 in response to an update data write request from an application. The module 201 writes update data to the address of the free area obtained from the controller 202 on the drive 101. The controller 202 detects a failure of the server 102 through the adapter 204 and registers the failed server in a failed server list. When the failed server 102 outputs write completion notification to the controller 202, the controller 202 does not register the area written by the server 102 in a file index because the server 102 has been registered in the failed server list.

INPADOC Legal Status: None **Buy Now:** Family Legal Status Report

Family: Show 3 known family members

Other Abstract Info: DERABS G2001-195468 DERABS G2001-195468

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（A）　(11)特許出願公開番号
特開2001－14201
(P2001－14201A)
(43)公開日　平成13年１月19日(2001.1.19)

(51)Int.Cl.'		識別記号	FI		テーマコード'(参考)
G06F	12/00	535	G06F	12/00	535Z　5B045
	3/06	301		3/06	301C　5B065
	11/00	330		11/00	330A　5B082
	15/177	682		15/177	682J

審査請求　未請求　請求項の数６　ＯＬ　（全７頁）

(21)出願番号　特願平11－183218

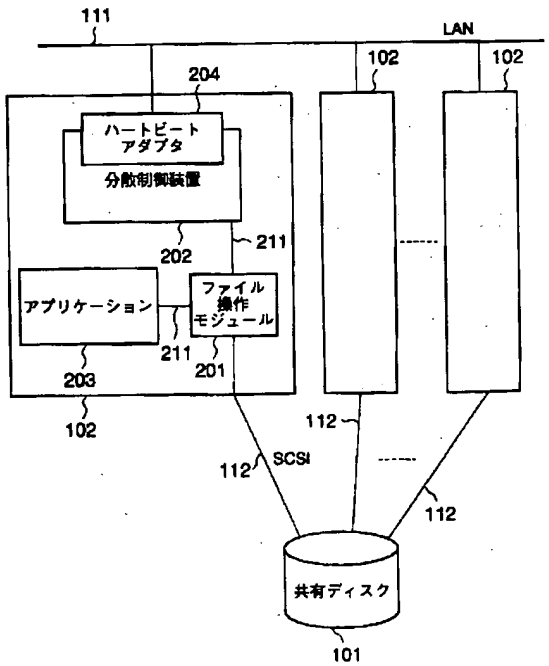
(22)出願日　平成11年６月29日(1999.6.29)

(71)出願人　000003078
株式会社東芝
神奈川県川崎市幸区堀川町72番地
(72)発明者　山本　俊行
東京都府中市東芝町１番地　株式会社東芝
府中工場内
(74)代理人　100058479
弁理士　鈴江　武彦　（外６名）

Ｆターム(参考)　5B045 DD01 EE06 GG01 JJ09
5B065 BA01 CA02 CC03 ZA08
5B082 CA08 DA01 DC02 DE01 FA16
GB06

(54)【発明の名称】　共有ファイルシステム

(57)【要約】
【課題】スプリットブレインが起こっても、ファイルシステムを破壊しない共有システムを提供すること。
【解決手段】各サーバ１０２には分散制御装置２０２、ファイル操作モジュール２０１およびハートビートアダプタ２０４が設けられる。アプリケーションからの更新データ書き込み要求をに対して、分散制御装置２０２は、共有ディスク装置１０１上の空き領域の番地をファイル操作モジュール２０１に出力する。ファイル操作モジュール２０１は、共有ディスク装置１０１上の前記分散制御装置２０２から得た空き領域の番地に更新データを書き込む。分散制御装置２０２は、ハートビートアダプタ２０４を介して、サーバ１０２の故障を検知し、故障サーバリスト３０４に故障したサーバを登録する。この故障したサーバ１０２から書き込み完了通知が分散制御装置２０２に出されると、分散制御装置２０２は、このサーバ１０２が故障サーバリスト３０４に登録されているため、このサーバ１０２が書き込んだ領域をファイルインデックス４０１に登録しない。



【特許請求の範囲】
【請求項１】　ネットワークを介して接続された複数のコンピュータと、前記複数のコンピュータとバスを介して接続された共有ディスク装置とを有する共有ファイルシステムにおいて、
任意のファイルが格納される前記共有ディスク装置上の位置を記録するファイルインデックスと、前記共有ディスク装置上の空き領域を記録する空き領域リストとを少なくとも有し、アプリケーションからの前記共有ディスク装置への更新データの書き込み要求に対し、前記空き領域リストを参照して前記共有ディスク装置上の空き領域の番地を出力する分散制御装置と、
前記分散制御装置から出力された空き領域の番地に基づいて前記共有ディスク装置に更新データを書き込むファイル操作モジュールと、
を具備したことを特徴とする共有ファイルシステム。
【請求項２】　前記各コンピュータは、各々他のコンピュータの故障を検知するためのハートビートアダプタを有し、
前記分散制御装置はさらに故障したコンピュータを記録する故障コンピュータリストを更に有し、
前記分散制御装置は、前記ハートビートアダプタにより前記コンピュータの故障を検知すると、故障したコンピュータを示す情報を前記故障サーバリストに登録することを特徴とする請求項１記載の共有ファイルシステム。
【請求項３】　前記ファイル操作モジュールは、アプリケーションからの要求に応答して、前記分散制御装置に対してコンピュータ同定情報を引数として空き領域取得要求を出力し、
前記分散制御装置は、前記空き領域取得要求に対し、前記取得コンピュータリストに、前記引数で与えられた故障コンピュータが登録されているか否かを調べ、登録されている場合には、エラー情報を前記ファイル操作モジュールに返し、登録されていない場合には、前記空き領域を参照して空き領域の番地を前記ファイル操作モジュールに返すことを特徴とする請求項２記載のファイル共有システム。
【請求項４】　前記ファイル操作モジュールは、アプリケーションからの書き込み要求に対して、前記共有ディスク装置上の空き領域に書き込みを行ない、書き込みが完了すると、サーバ情報を引数として書き込み完了通知を前記分散制御装置に出力し、
前記分散制御装置は、前記故障コンピュータリストを参照して、前記引数で与えられたコンピュータが登録されているか否か調べ、登録されている場合には、その故障コンピュータが書き込んだ前記共有ディスク装置上の書き込み領域を示す位置情報を前記ファイルインデックスに登録せず、エラー情報を前記ファイル操作モジュールに返し、登録されていない場合には、書き込んだ領域を示す共有ディスク装置上の位置を前記ファイルインデックスに登録することを特徴とする請求項２記載のファイル共有システム。
【請求項５】　前記分散制御装置は、前記書き込んだ領域を前記ファイルインデックスに登録した後、さらに、更新前のデータが書き込まれている前記共有ディスク装置上の領域を前記空き領域リストに登録することを特徴とする請求項２記載のファイル共有システム。
【請求項６】　前記ファイル操作モジュールは、アプリケーションからの読み込み要求に応答して、前記分散制御装置にコンピュータ同定情報、ファイル番号、およびファイル位置情報を引数として番地取得要求を前記分散制御装置に出力し、
前記分散制御装置は、前記番地取得要求に応答して、前記故障コンピュータリストに、前記引数で与えられたコンピュータが登録されているか否か調べ、登録されている場合には、エラー情報を前記ファイル操作モジュールに返し、登録されていない場合には、前記ファイルインデックスを参照し、引数として与えられたファイル番号のファイル位置の内容が書かれている前記共有ディスク装置上の番地を前記ファイル操作モジュールに返すことを特徴とする請求項２記載のファイル共有システム。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】この発明は、共有ファイルシステムに関し、特に分散環境における共有ファイルシステムに関する。
【０００２】
【従来の技術】近年、分散システムは急速に進歩しつつある。分散システムは、１）負荷分散に基づく並列処理による処理能率および応答時間特性の改善、２）負荷の機能分散による処理能率および応答時間特性の改善、３）複数の計算機が利用可能なことによる信頼性および可用性の増大、４）負荷の増大に対し、計算機の増設で容易に対処できる高い機能拡張性等の利点があり、特に地理的分散システムでは、５）データの処理要求が発生した地点（サイト）で処理を行えるため、通信回線コストの低減と応答時間特性の改善も可能になる。
【０００３】このような分散制御システムとして、ネットワークファイルシステムが知られている。ネットワークファイルシステムは、ＬＡＮ(Local Area Network)上でファイルを共有するシステムである。このようなネットワークシステムにおいては、ＬＡＮ上に接続された複数のサーバのうちのいずれか１台がマスタサーバとして、共有ディスクをアクセスできるように構成されている。従って、他のサーバが共有ディスクをアクセスする場合には、マスタサーバを介してデータの授受が行われる。
【０００４】ところで、従来のファイルシステムは、１）データ領域および２）ファイルアロケーションテーブル（ＦＡＴ）の２つの要素から構成される。データ領

域は、ファイルの内容を記録する領域であり、数Kバイト毎に区切られたブロックで構成され、各ブロックには番地がつけられてる。また、ＦＡＴは、目的のファイルがデータ領域のどの番地に記録されているかを記録するテーブルである。あるファイルからデータを読み込む際には、ＦＡＴを参照し、実際のデータがある番地を取得して、その番地のデータを読取る。同様に、あるファイルにデータを書き込む際には、ＦＡＴを参照し、データを書き込む番地を取得して、その番地にデータを書き込む。

【０００５】

【発明が解決しようとする課題】このようなファイルシステムにおいては、スプリットブレインが起きたとき、ファイルが壊されてしまう恐れがある。スプリットブレインとは、ネットワークにより同期を取って動いていた複数台のコンピュータが、ネットワークの故障や高負荷により同期が取れなくなり、２つの組に別れてしまうことを言う。このような状態になったとき、お互いの組が同じファイルをアクセスすると、一方のコンピュータが書き換えた内容を他方のコンピュータが書き換えてしまい、データが破壊されてしまうという問題が起きる。

【０００６】この発明は、上記事情に鑑みてなされたもので、その目的は、スプリットブレインが起こってもファイルシステムを破壊しないようにした共有ファイルシステムを提供することである。

【０００７】

【課題を解決するための手段】この発明によれば、ネットワークを介して接続された複数のコンピュータと、前記複数のコンピュータとバスを介して接続された共有ディスク装置とを有する共有ファイルシステムにおいて、任意のファイルが格納される前記共有ディスク装置上の位置を記録するファイルインデックスと、前記共有ディスク装置上の空き領域を記録する空き領域リストとを少なくとも有し、アプリケーションからの前記共有ディスク装置への更新データの書き込み要求に対し、前記空き領域リストを参照して前記共有ディスク装置上の空き領域の番地を出力する分散制御装置と、前記分散制御装置から出力された空き領域の番地に基づいて前記共有ディスク装置に更新データを書き込むファイル操作モジュールとを具備したことを特徴とする。

【０００８】この発明によれば、アプリケーションから更新データの書き込み要求がファイル操作モジュールに出力される。ファイル操作モジュールは、サーバ番号を引数として空き領域取得要求を分散制御装置に出力する。分散制御装置は、サーバ番号を引数として、故障サーバリストを参照し、引数で示されるサーバが故障サーバリストに登録されているか否かチェックする。登録されていれば、分散制御装置は、エラー情報をファイル操作モジュールに返す。この結果、更新データは共有ディスク装置に書き込まれない。一方、故障サーバリストに登録されていなければ、空き領域リストを参照して、共有ディスク装置上の空き領域の番地をファイル操作モジュールに返す。この結果、ファイル操作モジュールは、分散制御装置から得た番地に更新データを書き込む。このように、この発明によれば、更新データは、更新前のデータが書かれた領域に上書きするのではなく、共有ディスク装置上の空き領域に順次書き込むように構成したので、例え、スプリットブレインが起こってもファイルを破壊する恐れが無い。しかも、更新データの書き込み完了後、ファイル操作モジュールは、書き込み完了通知を、分散制御装置に送り、分散制御装置はその段階においても、故障サーバからの書き込み完了通知か否かを故障サーバリストを参照してチェックし、故障サーバからの書き込み完了通知である場合には、書き込んだ領域をファイルインデックスに登録しないので、その領域は次に再利用される領域とはなり得ないので、スプリットブレインを起こしたサーバが書き込む領域に、他のサーバが書き込みを行う可能性はない。従って、ファイルの破壊を防止することができる。

【０００９】また、正常なサーバからの書き込み完了通知を受けた場合には、分散制御装置は、更新前のデータが書き込まれている領域を新たな空き領域として、空き領域リストに登録し、再利用を図るので、空き領域が不足するという恐れはない。

【００１０】

【発明の実施の形態】以下、図面を参照して本発明の実施形態を説明する。

【００１１】図１は、この発明の共有ファイルシステムの一実施形態を示すブロック図である。同図において、複数のコンピュータ（以下、サーバと呼ぶ）１０２が例えばEthernet等のローカルエリアネットワーク（ＬＡＮ）を介して接続されている。さらに、これらの複数のサーバ１０２は、例えばＳＣＳＩ(Small Computer System Interface)のバス１１２を介して共有ディスク装置１０１に接続されている。各サーバ１０２はそれぞれ独立して、共有ディスク装置１０１をアクセスすることができる。但し、複数のサーバ１０２が同時に共有ディスク装置１０１をアクセスすることはできないので、排他制御が行われる。この排他制御は各サーバ１０２内に組込まれた後述する分散制御装置２０２により制御される。

【００１２】図２はサーバ１０２の内部構成を示すブロック図である。図２に示すように、各サーバ１０２は、ハートビートアダプタ２０４、分散制御装置２０２、分散制御装置２０２と内部ネットワーク２１１を介して接続されたファイル操作モジュール２０１およびファイル操作モジュール２０１と内部ネットワーク２１１を介して接続されたアプリケーション２０３を有する。なお、図面の簡単のために、１台のサーバ１０２についてのみその内部構成を示しているが、他のサーバ１０２も同様

の構成を有する。

【0013】ハートビートアダプタ204は各サーバ102の対応するハートビートアダプタ204と通信し合うことにより、各サーバ102が互いに連携してあたかも1つの装置であるかのように動作することができる。このハートビートアダプタ204により、通信できなくなったサーバ102は、故障と判断される。

【0014】分散制御装置202は、複数台のサーバ102を連携する装置である。各サーバ102の分散制御装置202は、ハートビートアダプタ204により、互いに通信し合い、あたかも1つの装置であるかのように動作する。すなわち、分散制御装置202上におかれる情報はもちろん、実行されるプログラムについても1つの装置202上で実行しているかのように動作する。分散制御装置202は、例えば、ネットワーク環境において、分散ノードを連携し、高信頼、高可用、かつスケーラブルなシステムを構築するためのミドルウエアで構成される。各サーバ102は、多重化されているので、どのサーバ102が故障しても、共有ファイルシステムは全体として動作しつづける。また、前記ハートビートアダプタ204により、連動しているサーバ102の故障を検知するので、障害に対する対処が容易に行える。

【0015】図3に分散制御装置202の内部構成を示す。

【0016】同図において、分散制御装置202上は、共通制御モジュール301、およびそれぞれ内部ネットワーク211を介して接続された、ファイルインデックス302、空き領域リスト303、および故障サーバリスト304を有する。共通制御モジュール301は、すべてのサーバ102で同じ動作をする。ファイルインデックス302は、任意のファイルが共有ディスク装置101上のどの位置に格納されているかを記録する。ファイルインデックス302は複数のインデックスリストから構成され、各インデックスリストはファイル毎に作られる。

【0017】共有ディスク装置101は、図5に示すように例えば、4Kバイト毎のデータブロック502に分割されている。各データブロック502には、それぞれディスク番地501が割り付けられている。今、図4（a）に示すように、ファイルインデックス302の中の符号401で示されるファイル番号が「1」で、かつ符号402で示されるファイル位置が「1」のデータが「11」である場合、図5において、符号501で示すディスク番地が「11」番地のデータブロック502にファイルが記録されているということを表す。

【0018】空き領域リスト303は、共有ディスク装置101上のどの領域が空き領域かを記録するもので、図4（b）に示すように、ファイルインデックス302と同様、ディスク番地501を記録する。また、故障サーバリスト304は、故障したサーバを記録するもので、図4（c）に示すように、各サーバにあらかじめ与えられたサーバ番号403を記録する。

【0019】図2のアプリケーション203は、共有ディスク装置101に格納されたファイルに対するアクセスを行うプログラムである。ファイル操作モジュール201は、アプリケーション203からの読み込みおよび書き込み要求に応答して、分散制御装置202と連携を取りながら、共有ディスク装置101をアクセスするモジュールである。

【0020】以下、この発明の一実施形態の動作について説明する。

【0021】アプリケーション203は、ファイル操作モジュール201に対してファイルへの読み込みおよび書き込みを要求する。すなわち、読み出し動作の場合、アプリケーション203は、引数として、1）読み込むファイルの番号および2）ファイル内の読み込む位置を設定し、ファイル操作モジュール201に読み込みを要求する。これに対し、ファイル操作モジュール201は、サーバ番号、読み込むファイルの番号および読み込む位置を引数として、分散制御装置202にディスク番地取得要求を出力する。分散制御装置202の共通制御モジュール301は、ディスク番地取得要求に応答して、まず、サーバ番号を引数として故障サーバリスト304を参照して、引数で示されるサーバ番号が記録されているか否かチェックする。記録されている場合には、そのサーバは故障していると他のサーバ102に認識されているので、アクセスを認めると共有ディスク装置101上のデータが破壊される恐れがある。このため、共通制御モジュール301は、アクセスエラーを示す情報をファイル操作モジュール201に返す。一方、引数で示されるサーバ番号が故障サーバリスト304に記録されていなければ、共通制御モジュール301は、ファイルインデックス302を参照し、引数で指定されたファイル番号で示されるファイルの指定された読み込み位置がどのディスク番地に書き込まれているかを調べ、そのディスク番地をファイル操作モジュール201に返す。ファイル操作モジュール201は、共有ディスク装置101をリードアクセスし、分散制御装置202から得たディスク番地からファイルデータをリードし、戻り値として、アプリケーション203に出力する。

【0022】一方、書き込み動作の場合、アプリケーション203は、引数として1）書き込むファイルの番号、2）書き込む位置、および3）書き込むデータを指定し、ファイル操作モジュール201に書き込み要求を行う。

【0023】ファイル操作モジュール201は、サーバ番号を引数として、分散制御装置202に対して空き領域取得要求を出力する。分散制御装置202の共通制御モジュール301は空き領域取得要求に応答して、まず、サーバ番号を引数として、故障サーバリスト304

を参照し、引数で示されるサーバ番号が記録されているか否か調べる。記録されていれば、上述したようにエラー情報をファイル操作モジュール２０１に返す。

【００２４】一方、引数で示されるサーバ番号が、サーバ故障リスト３０４に記録されていなければ、次に、空き領域リスト３０３を参照し、空き領域のディスク番地を得た後、空き領域リストからこの空き領域を削除し、空き領域のディスク番地をファイル操作モジュール２０１に出力する。ファイル操作モジュール２０１は、共有ディスク装置１０１にアクセスし、共通制御モジュール３０１から得た空き領域のディスク番地に目的のデータを書き込み、書き込みを完了すると、書き込み完了通知を共通制御モジュール３０１に出力する。

【００２５】共通制御モジュール３０１は、書き込み完了通知に応答して、サーバ番号を引数として故障サーバリスト３０４を参照し、引数のサーバ番号が故障サーバリスト３０４に記録されているか否かチェックする。記録されている場合には、上述したように、エラーを示す情報をファイル操作モジュール２０１に返す。一方、故障サーバリスト３０４に記録されていなければ、次に、共通制御モジュール３０１は、引数で示されるファイル番号のファイル位置の内容が引数で示されるディスク番地に格納されていることをファイルインデックス３０２に登録する。そして、共通制御モジュール３０１は、元のディスク番地、すなわち更新前のデータが書き込まれていたディスク番地を空き領域リスト３０３に追加する。そして、共通制御モジュール３０１は、書き込み完了通知に対する処理が完了したことを示す情報をファイル操作モジュール２０１に返す。

【００２６】ファイル操作とは非同期に、分散制御装置２０２は常に各サーバ１０２が動作しているかをハートビートアダプタ２０４により監視している。ハートビートアダプタ２０４は、各サーバ１０２に信号（ハートビート）を出すことにより、自分が動作していることを示し、それぞれのサーバから信号（ハートビート）を受け取ることにより、そのサーバが動作していることを知る。サーバからの信号が途絶え、そのサーバが故障したと判断したときは、そのサーバ（サーバ番号）を故障サーバリスト３０４に追加する。

【００２７】いま、スプリットブレインが起こり、あるサーバ１０２がある共有ディスク装置１０１のある領域にファイルデータを書き込もうとした瞬間に一時停止し、このサーバがいつ再始動し、この領域に書き込みをするかはわからないとする。一時停止しているサーバ１０２はハートビートを出せなくなるので、分散制御装置２０２は、そのサーバが故障したと判断し、故障サーバリスト３０４に追加し、以降このサーバからの要求は受け付けられない。他のサーバ１０２は平常通りファイルへ読み書きを行う。もちろん、一時停止しているサーバ１０２が書き込もうとしているファイルに対しても読み書きを行う。この状態において、一時停止していたサーバ１０２が再始動して、共有ディスク装置１０１に書き込みを行ったとする。このサーバ１０２が書き込んだ領域は、まだ空き領域リスト３０３に追加されていないので、他のサーバ１０２が、この領域にデータを書き込んでいる可能性はない。すなわち、上述したように、このサーバ１０２が空き領域取得要求を出した時点で、この領域は空き領域リスト３０３から削除されているので、他のサーバがデータを書き込むために、空き領域取得要求を出しても、この領域を取得する可能性は無い。書き込みが完了し、分散制御装置２０２に書き込み番地を返しても、このサーバはすでに故障サーバリスト３０４に登録されているため、この要求は受け付けられない。そのため、ファイルインデックス３０２にこの領域が登録されることはない。すなわち、ファイルが書き込まれているデータを書き換える可能性は無く、またファイルインデックス３０２を書き換える可能性も無い。

【００２８】なお、ファイルに対する更新データは、元のデータに上書きするのではなく、ディスクの空き領域に書き込まれるため、ディスク領域が足りなくなるのではないかという懸念が生じるが、更新データを空き領域に書き込み後、更新前のデータが書かれていた領域は、空き領域リスト３０３に加えられ、空き領域として再利用される。このため、使用中のディスク領域は、実際にデータが入っている領域と、現在書き込み中の領域のみであり、利用できるディスク領域が足りなくなるということはない。

【００２９】

【発明の効果】この発明によれば、分散制御装置２０２は、ハートビートアダプタ２０４によりサーバ１０２の故障を検知し、故障サーバリスト３０４に登録する。このため、仮に、サーバ１０２が、例えば過負荷等の何らかの原因で一時的に停止した状態となり、その後再始動し、空き領域にデータを書き込み、書き込み完了通知を分散制御装置２０２に出力しても、分散制御装置２０２、故障サーバリスト３０４にこのサーバ１０２が登録されているため、エラーをサーバ１０２に返す。従って、スプリットブレインが起こっても、ファイルシステムを破壊する恐れがない。

【図面の簡単な説明】

【図１】この発明の順次記録型共有ファイルシステムの一実施形態を示すシステムブロック図である。

【図２】図１に示すサーバの内部構成を示すブロック図である。

【図３】図２に示す分散制御装置の内部構成を示すブロック図である。

【図４】分散制御装置上に保持される情報を示し、（ａ）は、ファイルインデックスを、（ｂ）は空き領域リストを、（ｃ）は故障サーバリストをそれぞれ示す。

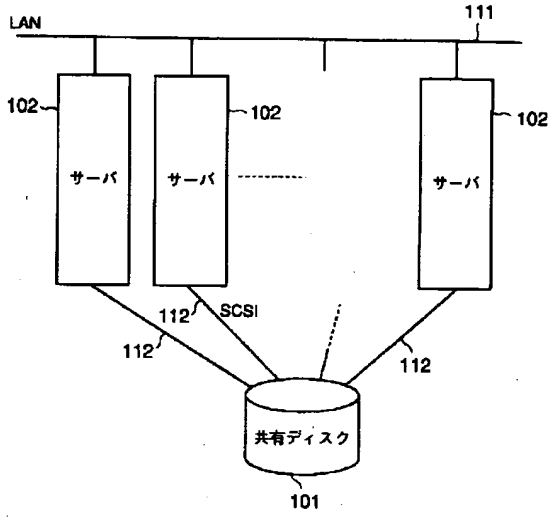
【図５】図１に示す共有ディスク装置１０１の領域の構

成を示す概念図である。

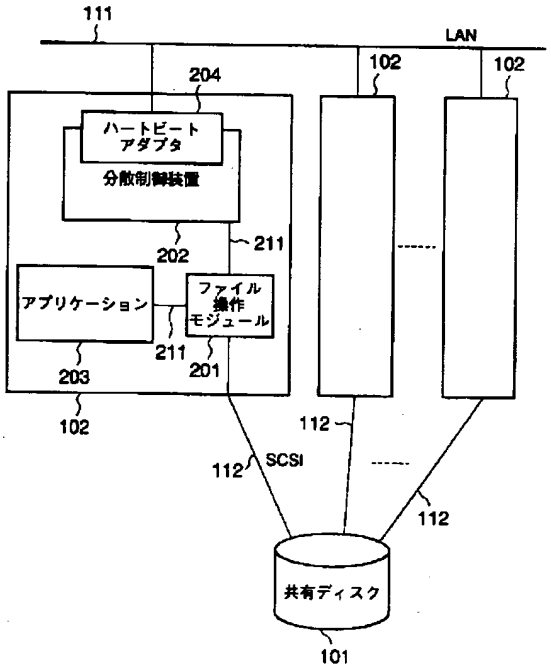
【符号の説明】

101…共有ディスク装置
102…サーバ
111…LAN
112…SCSI
201…ファイル操作モジュール
202…分散制御装置
203…アプリケーション
204…ハートビートアダプタ
211…内部ネットワーク
301…共通制御モジュール
302…ファイルインデックス
303…空き領域リスト
304…故障サーバリスト
401…ファイル番号
402…ファイル位置
403…サーバ番号
501…ディスク番地
502…データブロック

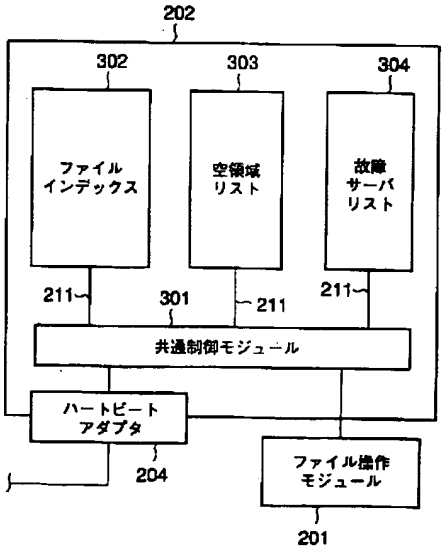
【図1】



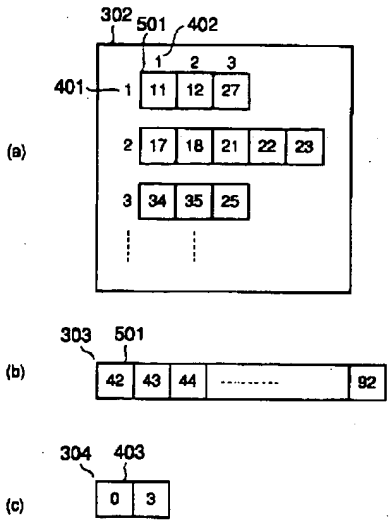
【図2】



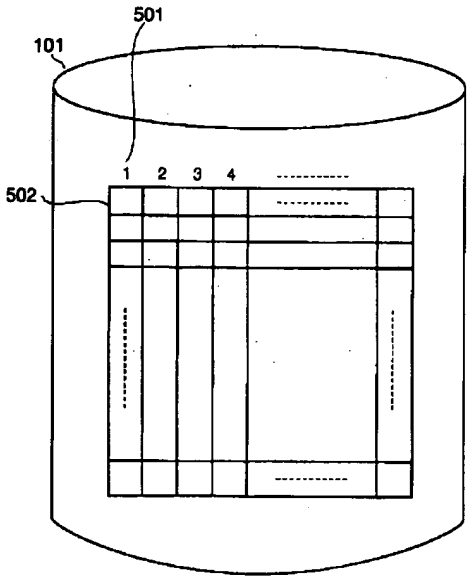
【図3】



【図4】


302
501
402
1
2
3
401
1
11
12
27
(a)
2
17
18
21
22
23
3
34
35
25
303
501
(b)
42
43
44
92
304
403
(c)
0
3

【図5】


501
101
1
2
3
4
502